

**An Investigation of Using an RQP Based Method to Calculate  
Parameter Sensitivity Derivatives**

**Todd J. Beltracchi  
Graduate Student**

**Gary A. Gabriele  
Assistant Professor**

**Department of Mechanical Engineering,  
Aeronautical Engineering & Mechanics**

**Rensselaer Polytechnic Institute  
Troy, N.Y. 12180-3590**

## **Parameter Sensitivity Analysis**

Estimation of the sensitivity of problem functions with respect to problem variables forms the basis for many of our modern day algorithms for engineering optimization. The most common application of problem sensitivities has been in the calculation of objective function and constraint partial derivatives for determining search directions and optimality conditions. A second form of sensitivity analysis, parameter sensitivity, has also become an important topic in recent years. By parameter sensitivity, we refer to the estimation of changes in the modeling functions and current design point due to small changes in the fixed parameters of the formulation. Methods for calculating these derivatives have been proposed by several authors (Armacost and Fiacco 1974, Sobieski et al 1981, Schmit and Chang 1984, and Vanderplaats and Yoshida 1985). Two drawbacks to estimating parameter sensitivities by current methods have been: (1) the need for second order information about the Lagrangian at the current point, and (2) the estimates assume no change in the active set of constraints. This paper addresses the first of these two problems and proposes a new algorithm that does not require explicit calculation of second order information.

**The estimation of changes in the modeling functions and design point due to small changes in the fixed parameters of the formulation.**

## Standard Form of NLP Parameter Sensitivity Problem

To provide a framework about which to address the problem of parameter sensitivity analysis, we state the following standard form of the nonlinear programming problem which explicitly represents the problem parameters.

In the formulation given, we assume that the problem functions  $f$ ,  $g$ , and  $h$  can be either linear or nonlinear functions of the design variables but we are concerned primarily with the nonlinear case. We assume that the problem parameters  $p$ , are held fixed during the course of the optimization, and the optimal solution point,  $x^*$ , satisfies the first order Kuhn-Tucker optimality conditions.

$$\begin{array}{ll}\text{Minimize} & f(x, p) \\ \text{Subject to} & h_l(x, p) = 0 \quad l = 1, L \\ & g_j(x, p) \geq 0 \quad j = 1, J \\ & x_{\min} \leq x \leq x_{\max} \\ & x = (x_1, x_2, \dots, x_n) \\ & p = (p_1, p_2, \dots, p_k)\end{array}$$

**OBJECTIVE:** For a given  $p$ , find  $x^*$ , that satisfies the above problem. We are then interested in the effects of variations in  $p$  on the optimum.

**Given:**  $p_i' = p_i + \Delta p_i$

**Find:**  $f(x^*, p')$ ,  $x_{\text{new}}^*$

## Required Formulas

For any change in the parameter  $\Delta p_i$ , the new optimum value of the objective function or design variables can be estimated from the following linear extrapolations:

Extrapolations based on these equations are bounded by the assumption that the active set remains the same.

$$f(x^*, p') = f(x^*_{old}) + \Delta p_i \frac{df^*}{dp_i}$$

$$x^*_{new} = x^*_{old} + \Delta p_i \frac{\partial x^*}{\partial p_i}$$

where

$$\frac{df^*}{dp_i} = \frac{\partial f}{\partial p_i} + \sum_{l=1}^L v_l \frac{\partial h_l}{\partial p_i} - \sum_{j=1}^J u_j \frac{\partial g_j}{\partial p_i}$$

$$\frac{df^*}{dp_i} = \frac{\partial f}{\partial p_i} + \frac{\partial f^T}{\partial x} \frac{\partial x^*}{\partial p_i}$$

**Derivatives to be determined:**

$$\frac{\partial f}{\partial p_i}, \quad \frac{\partial x^*}{\partial p_i}, \quad \frac{\partial h_l}{\partial p_i}, \quad \frac{\partial g_j}{\partial p_i}$$

## Methods for Calculating Parameter Sensitivities

### The Brute Force Method

The brute force method is probably the most common method used to study the effect of problem parameters on solutions. The method is simply to change the parameter and then reoptimize the problem with the new value. This of course gives the truest indication of the effect of the parameter on the solution. A variation of the brute force method was proposed by Armacost and Fiacco (1974) and McKeown (1980) to calculate parameter sensitivities based on the central difference approximation given below.

Given the incremental change  $\Delta$  in  $p_i$ , reoptimize the original problem at the new value of  $p$ . The sensitivity derivatives are given by the difference formulas.

$$\frac{df^*}{dp_i} = \frac{f(x^*, p_i + \Delta p_i) - f(x^*, p_i - \Delta p_i)}{2\Delta p_i}$$

$$\frac{\partial x^*}{\partial p_i} = \frac{x^*(p_i + \Delta p_i) - x^*(p_i - \Delta p_i)}{2\Delta p_i}$$

## Methods for Calculating Parameter Sensitivities

### Kuhn-Tucker Method

A more accurate estimate of the sensitivity derivatives can be found by differentiating the Kuhn-Tucker optimality conditions with respect to a parameter. We refer to this as the Kuhn-Tucker method. The set of Kuhn-Tucker sensitivity equations have been derived independently by several authors (Armacost and Fiacco 1974, McKeown 1980, Sobieski et al. 1981) and result in the following linear system of equations.

**Differentiate the Kuhn-Tucker conditions wrt to  $p_i$  and solve the resulting linear system for the desired derivatives.**

$$\begin{bmatrix} \nabla_x^2 L & -\nabla_x(h,g) \\ \nabla_x(h,g)^T & 0 \end{bmatrix} \begin{bmatrix} \frac{\partial x^*}{\partial p_i} \\ \frac{\partial(v,u)}{\partial p_i} \end{bmatrix} + \begin{bmatrix} \frac{\partial \nabla_x L}{\partial p_i} \\ \frac{\partial(h,g)}{\partial p_i} \end{bmatrix} = 0$$

# Methods for Calculating Parameter Sensitivities

## Extended Design Space Method

The final category of parameter sensitivity methods has been proposed by Vanderplaats (1984,1987). The method, known as the extended design space (EDS) method, is based on using feasible directions for estimating parameter sensitivity derivatives. The method extends the design space and solves the following subproblem to obtain the sensitivity derivatives. Both first and second order estimates have been developed for the method (Vanderplaats and Yoshida 1985).

The fixed parameter  $p_i$  is added to the set of design variables,

$$x_{n+1} = p_i$$

Solve the following subproblem for  $s$ ,

$$\begin{aligned} \min \quad & \nabla_x f^T s \\ \text{st.} \quad & g_j + \nabla_x g_j^T s \geq 0 \quad j=1,J \end{aligned}$$

Calculate desired sensitivities from

$$\frac{\partial x}{\partial p_i} = (s_1, s_2, \dots, s_n) / s_{n+1}$$

$$\frac{df}{dp} = \frac{\partial f}{\partial p} + \frac{\partial f^T}{\partial x} \frac{\partial x}{\partial p}$$

## Assessment of Current Methods for Calculating Parameter Sensitivities

As evidenced by their lack of extensive use, all the methods discussed above have some drawback associated with their use. Because the problem has to be reoptimized for several different values of the parameters, the efficiency of the Brute Force method is affected by the difficulty of the problem and the efficiency of the method used in the reoptimization. This approach is useful in studying large variations in parameters by plotting the response of the optimum versus the parameter, and has been used by Arbuckle and Sliwa (1984) and Robertson and Gabriele (1987).

The Kuhn-Tucker method is also computationally expensive because it requires second derivatives of the objective function and the active constraints. For most engineering design problems, this type of information may be difficult to obtain. This method requires that the strict complementarity and linear independence assumptions hold at the optimal design.

Finally, the first order ESD method is a very efficient, easy to implement method but it can provide inaccurate estimates of  $\partial x^*/\partial p$  when the problem is not fully constrained and it does not provide  $\partial u/\partial p_i$ . The second order EDS method requires the calculation of second derivatives and also requires the solution of a quadratic approximating problem for each value of the parameter that is studied. However, the second order EDS method has the advantage of not being affected by changes in the active set.

**Brute Force Method:** Most commonly used method, provides accurate results, but inefficient.

**Kuhn-Tucker Method:** Sound mathematical basis, but assumes no changes in the active set and requires second order information.

**Extended Design Space:** Very efficient, easy to implement, but may not produce accurate estimates of  $\partial x/\partial p$  and does not provide  $\partial u/\partial p$



## **A Proposal for a New Method**

From the previous discussion, we can deduce that what is needed to improve current methods for parameter sensitivity analysis is an algorithm that does not require second derivatives, is able to accurately predict the sensitivity derivatives, and can calculate sensitivities at degenerate points. In this paper, we propose using a new algorithm based on the Recursive Quadratic Programming (RQP) method for accomplishing these goals.

Our reasoning for such a method is based on the following virtues of the RQP method. In terms of number of function evaluations, the RQP method appears to be one of the most efficient methods available. This has been demonstrated in any of the published comparison studies in which codes for these methods were participants (Schittkowski, 1980 and Belegundu and Arora, 1985). Although the method is sensitive to variable and objective function scaling, it is not sensitive to constraint scaling. Finally, the RQP method provides an estimate of the Hessian of the Lagrangian, which can be useful for other purposes, and it is very efficient at locating an optimum, when the starting point is close to the true optimum. Both of these last advantages will be exploited in the development of our method for sensitivity estimation based on the RQP method.

### **Proposal:**

**Employ the Recursive Quadratic Programming Method (RQP) in conjunction with the Brute Force Method to estimate the required derivatives.**

### **Reasoning:**

**The RQP method is very efficient when started near the optimum solution.**

**If the RQP method is used to solve the original problem, an approximation of the Hessian of the Lagrangian will be available.**

**Estimates of all derivatives, including  $\partial u / \partial p$  can be developed.**

## The Recursive Quadratic Programming Method

All RQP methods use the same basic strategy of linearizing the constraints and approximating the Hessian of the Lagrangian to form a quadratic programming (QP) subproblem. The QP subproblem is then solved for the search direction  $s$  and a new estimate of the Lagrange multipliers of the constraints. The search direction  $s$  is then used to calculate a new estimate of the optimum.

The step length  $\alpha$  is determined by minimizing a line search penalty function  $P$  of the general form given below, where  $\Omega$  represents some combination of the constraints and the Lagrange multipliers. The penalty function attempts to assure that both the objective function and the violation of the constraints are reduced. As the method converges, the step length  $\alpha$  which minimizes  $P(x,u,v,R)$  approaches 1.

Form the following subproblem to determine a search direction  $s$

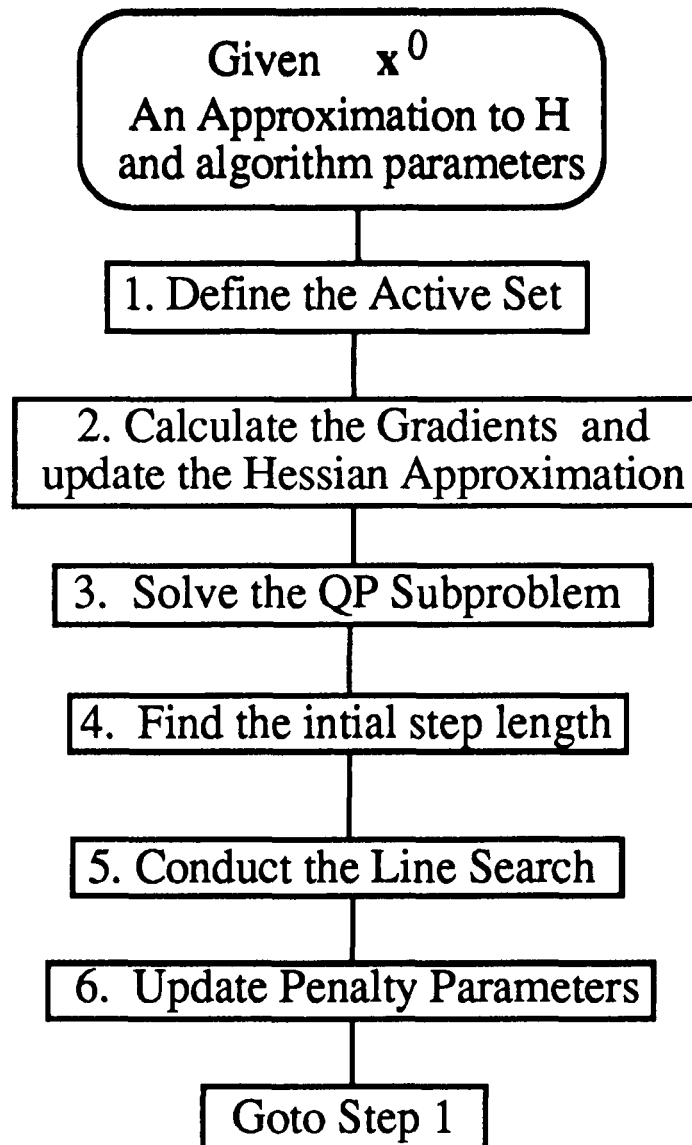
$$\begin{aligned} &\text{Minimize } 0.5 \ s^T B \ s + s^T \nabla f \\ &\text{subject to } \nabla h^T s + h = 0 \\ &\qquad \qquad \nabla g^T s + g \geq 0 \end{aligned}$$

Using  $s$ , perform a linear search to determine a new estimate of  $x^*$  by minimizing a penalty function of the following general form,

$$P(x,u,v,R) = f(x) + R \cdot \Omega(h,g,u,v)$$

## Basic Flowchart of RQP Method

The basic flowchart of the RQP method is given below. Several good implementations of the RQP method are available (Beltracchi and Gabriele 1987, Arora and Tseng 1987, Bartholomew-Biggs 1987, or Gill, et al. 1986). A more complete discussion of RQP methods can be found in (Beltracchi 1985 or Beltracchi and Gabriele 1988).



## The RQP Sensitivity Algorithm

The new algorithm is based on combining the simplicity of the brute force method with the efficiency of the RQP method. The two characteristics of the RQP method that we feel can be exploited for determining parameter sensitivities are (1) an approximation to the Hessian of the Lagrangian is developed, and (2) if this approximation is exact (or close) then the RQP method will quickly and efficiently solve the perturbed problem. In other words, if we can develop good Hessian approximations, the RQP method is equivalent to applying Newton's method to solve the Kuhn-Tucker conditions for the perturbed problem, which may require only 1 or 2 iterations of RQP. The small number of iterations, coupled with the fact that the RQP method should require only a one step line search, should allow the reoptimizations to occur without the need for many function evaluations.

The end result of combining the differencing equations and the RQP method is a means to estimate the parameter sensitivities without the need to calculate higher order derivatives, and without an excessive number of function evaluations. Based on the above arguments, we propose the following procedure to calculate parameter sensitivity derivatives.

- Step 0. Given an optimal solution  $x^*$ ,  $f^*$ ,  $u^*$ , an active set of constraints, and an approximation to the Hessian of the Lagrangian, all achieved by convergence of the RQP method (using the SR1/PD/BFS update).
- Step 1. Perturb the fixed parameter  $p_i$  to  $p_i^+ = p_i^0 + \Delta p_i$  where  $\Delta p_i$  is some small perturbation to  $p_i$ .
- Step 2. Perform two complete iterations of the RQP method. During the RQP iterations, update the Hessian approximation. From the results of the RQP iterations obtain  $f^+$ ,  $x^+$ ,  $u^+$ , and  $g_j^+$   $j \in$  active set.
- Step 3. Perturb the fixed parameter  $p_i$  to  $p_i^- = p_i^0 - \Delta p_i$ .
- Step 4. Start from  $x_0^- = x^*(p_i^0) - \frac{\partial x}{\partial p_i} \Delta p_i$  where  $\partial x / \partial p_i$  is calculated using forward differencing approximations and  $x^*(p_i^0)$  and  $x^+(p_i^0 + \Delta p)$  from step 2. Perform one complete iteration of the RQP method to find  $f^-$ ,  $x^-$ ,  $u^-$ , and  $g_j^-$   $j \in$  active set.
- Step 5. Obtain estimates for the sensitivity derivatives from the following central difference approximations

$$\frac{df^*}{dp} = \frac{f^+ - f^-}{2\Delta p}; \quad \frac{\partial x^*}{\partial p} = \frac{x^+ - x^-}{2\Delta p}$$

$$\frac{\partial u^*}{\partial p} = \frac{u^+ - u^-}{2\Delta p}$$

## Equivalence to the Kuhn-Tucker Method

The major questions to be answered about the proposed algorithm are does it provide the desired sensitivities, and what are the possible sources of error. An investigation of the theoretical properties of the RQP based sensitivity algorithm reveals that in the limit, as  $\Delta p$  goes to zero, the new method provides an estimate to the solution of the equations given below. These equations are equivalent to the Kuhn-Tucker sensitivity equations with the Hessian of the Lagrangian replaced by an approximation  $B$  that is provided by the RQP method. The details of this derivation are too lengthy to be presented here but are included in (Beltracchi 1988). From the above, we can see that the new algorithm will provide accurate estimates of the parameter sensitivities if  $B$  is a good approximation of  $\nabla_x^2 L$ , and the differencing formula is a good approximation of the equations below.

It can be shown that the proposed method is equivalent to the following linear set of equations,

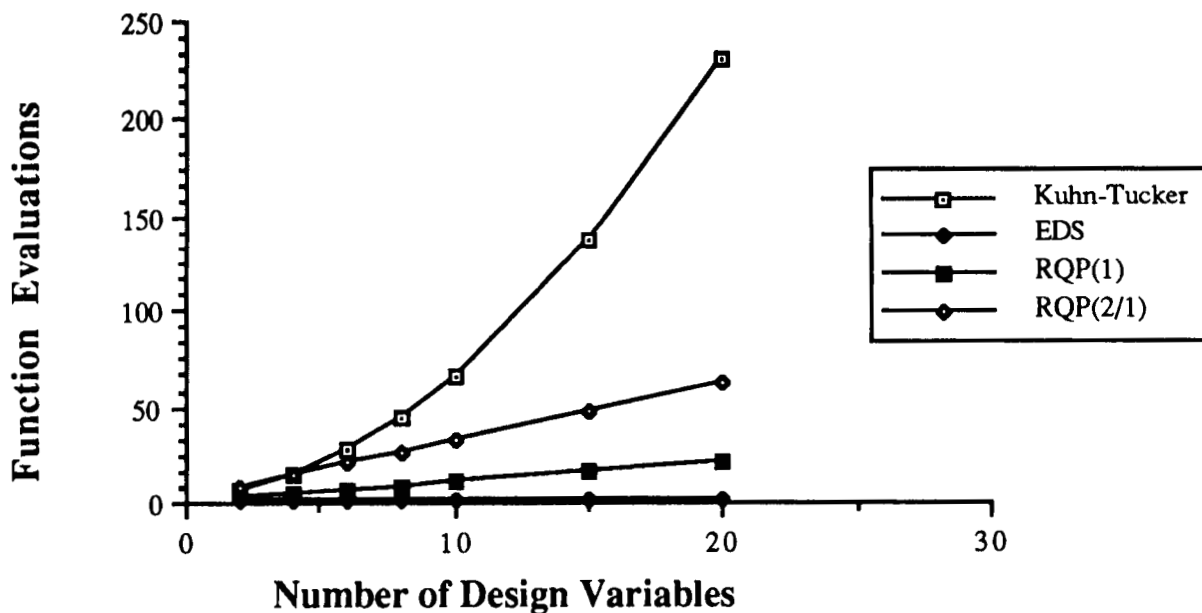
$$\begin{bmatrix} B & -\nabla_x g \\ \nabla_x g^T & 0 \end{bmatrix} \begin{bmatrix} \frac{\partial x}{\partial p} \\ \frac{\partial u}{\partial p} \end{bmatrix} + \begin{bmatrix} \frac{\partial \nabla_x L}{\partial p} \\ \frac{\partial g}{\partial p} \end{bmatrix} = 0$$

These equations are the Kuhn-Tucker equations with the Hessian Approximation  $B$  replacing  $\nabla_x^2 L$ .

## EFFICIENCY OF SENSITIVITY ALGORITHMS

As a measure of the efficiency of the new method, we present a comparison of the number of function evaluations required by the competing algorithms. The graph below represents the number of function evaluations required by various algorithms to find the sensitivity of the first parameter to be studied. The results are plotted for various problem sizes where  $n$  is the number of design variables.

Most of the work associated with the Kuhn-Tucker method is incurred when the Hessian matrix is calculated. For the first order EDS method, the work does not increase with problem size. However, as mentioned before, the method is not always accurate and  $\partial u/\partial p$  is not determined. Each of the RQP methods is either more efficient or as efficient when compared to the Kuhn-Tucker method for small problems ( $n < 5$ ), and considerably more efficient for larger problems ( $n > 5$ ).



## The SR1/PD/BFS Update

A major concern of the new algorithm is how well the Hessian approximation delivered from the RQP method agrees with the true Hessian. Toward this end, considerable research was conducted to find a variable metric update that provides good Hessian approximations without degrading the performance of the RQP algorithm. The two leading candidates were the Broyden-Fletcher-Shanno (BFS) method, and the Symmetric Rank One (SR1) method.

The SR1/PD/BFS (Symmetric Rank One/ Positive Definite/ Broyden Fletcher Shanno) is a hybrid variable metric update that combines the best features of the SR1 and BFS updates (Beltracchi 1988). The SR1 update has the advantage of not requiring exact line searches and producing good Hessian approximations, however it has the drawbacks of being undefined or producing indefinite Hessian approximations for some problems. The BFS update has the advantage of being self correcting; however, it has the drawback of requiring exact line searches, and the Hessian approximation does not converge unless fairly accurate line searches are performed. The SR1/PD/BFS update uses the BFS update when the SR1 update is undefined or likely to produce an indefinite Hessian approximation. The PD stands for a positive definite check (implemented in step 4), used to insure the new Hessian approximation is positive definite. Testing in Beltracchi (1988) found the SR1/PD/BFS update produced the best Hessian approximations.

(Symmetric Rank One / Positive Definite / Broyden Fletcher Shanno)

1. Calculate  $y^T(By - z)$
2. If  $\text{abs}(y^T(By - z)) \leq 10^{-10}$  goto step 7
3. Calculate  $\phi^*, \phi$ :

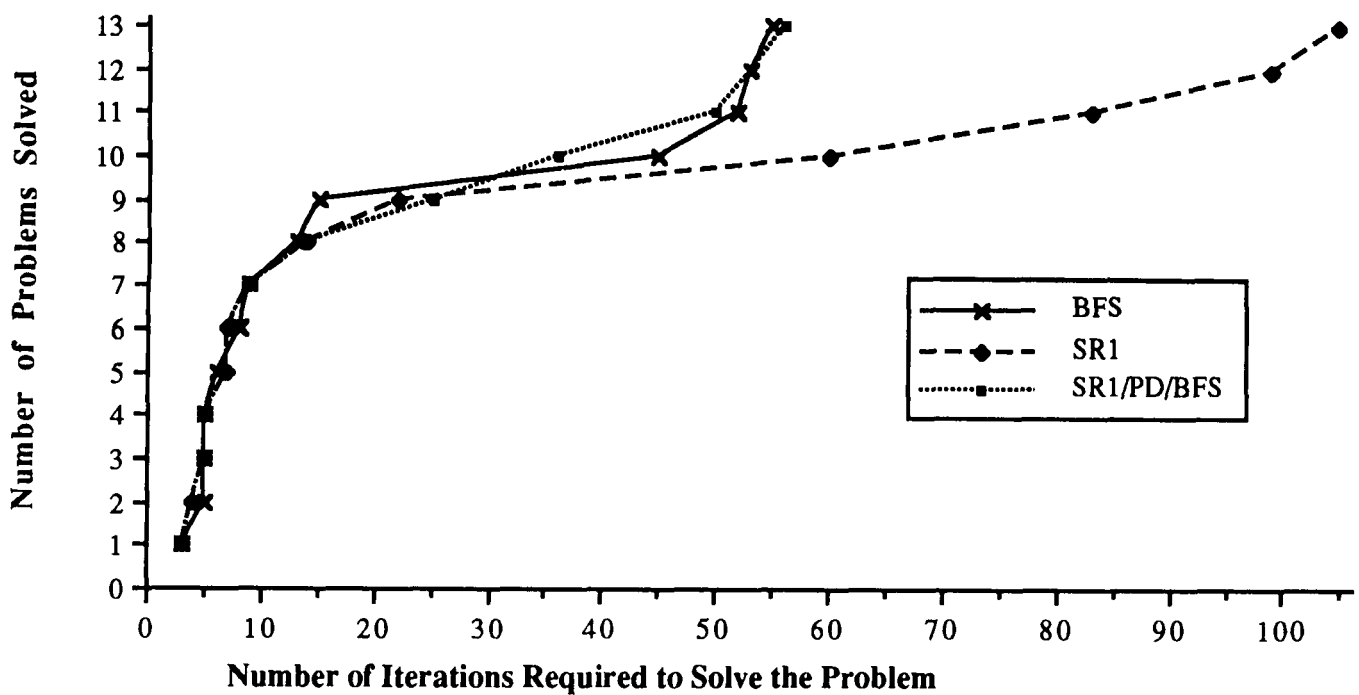
$$\phi^* = \frac{(y^T z)^2}{(y^T z)^2 - z^T B z \ y^T B^{-1} y}$$

$$\phi = \frac{y^T z}{y^T z - z^T B z}$$

4. If  $\phi \leq \phi^*$  goto step 7.
5. If  $y^T z \leq 0$  and  $y^T(By - z) \leq 0$  goto step 7.
6. Update Hessian approximation using the SR1 update and return.
7. Update Hessian approximation using the BFS update and return.

## PERFORMANCE OF NEW SR1/PD/BFS UPDATE

This slide shows the performance of the new SR1/PD/BFS variable metric update when implemented within the RQP method on a set of 13 commonly used test problems. The method is compared with the BFS and the SR1 method. We see from this plot that the new update provides the same level of robustness and efficiency as the BFS method and is generally more efficient than the SR1 method.





## Convergence of the Hessian Approximation for Various Updates in Broyden's Family

This table shows the ability of the tested variable metric updates to approximate the Hessian of the Lagrangian on a set of test problems. Analytical Hessians were developed for each test problem and compared to the approximated Hessian returned from the RQP method. The entries in the table report the Frobenius norms of the difference between the known optimal Hessian, and the identity matrix, and with the final approximate Hessian. From this we see that the new SR1/PD/BFS update returned approximations that were as good or better than the BFS method in all cases, and better than the SR1 method in all but one case. In this instance (Woods), the scaling of the objective function affected the final result. The last two rows demonstrate that scaling the objective allowed the new update to improve the approximation considerably.

Problem	$\ H^*-I\ _F$	BFS	SR1	SR1/PD/BFS
RTS01	2.291	1.396	0.061	0.0008
RTS01/100	1.377	0.0183	0.974	0.00023
RTS01*100	369.12	206.7	0.243	0.243
RTS02 st pt 1	7.384	3.00	3.0	3.0
RTS02 st pt 2	7.384	5.430	0.0	0.0
RTS03	15.93	7.76	0.105	0.105
RTS04	23.28	3.623	9.307	3.356
Woods	1352.46	141.3	17.17	75.38
Woods/100	13.54	1.908	0.184	0.0944
Woods/1000	1.394	0.0283	0.974	0.113

# TESTING OF RQP SENSITIVITY ALGORITHM

## INITIAL TEST SET

Two phases were employed in testing the new algorithm. The first phase involved a test set of known characteristics, whose Hessians and sensitivity derivatives could be determined analytically. The initial test set involved 4 test problems with 3-4 parameters each. Various algorithm parameters were studied, such as the step size  $\Delta p$ , as well as the effectiveness of the SR1/PD/BFS update, the number of RQP iterations to allow for solving the perturbed problem, and the updating of the Hessian approximation during the sensitivity analysis. The table below provides a sample of the results obtained on this initial test set using the RQP based sensitivity algorithm. The entries represent the error in the sensitivity derivatives obtained from the new algorithm and the known sensitivities. The errors were calculated using the formulas established by Sandgren (1977).

The results shown here were fairly typical of the results obtained on all the test problems in the initial test set. In general, the results are very good and certainly usable for engineering purposes.

	PROBLEM 1			PROBLEM 2	
	p 1	p 2	p 3	p 1	p 2
$\frac{df}{dp}$	.00002(10 <sup>-9</sup> )	1.9(10 <sup>-5</sup> )	0.126(10 <sup>-6</sup> )	0.139(10 <sup>-6</sup> )	0.00
$\epsilon_x$	0.00	7.9(10 <sup>-5</sup> )	3.11(10 <sup>-7</sup> )	9.08(10 <sup>-7</sup> )	2.00(10 <sup>-7</sup> )
$\epsilon_u$	0.00	2.66(10 <sup>-3</sup> )	1.55(10 <sup>-6</sup> )	9.79(10 <sup>-7</sup> )	1.43(10 <sup>-9</sup> )
$\epsilon_g$				6.33(10 <sup>-7</sup> )	0.00

# **TESTING OF RQP SENSITIVITY ALGORITHM**

## **ENGINEERING TEST PROBLEMS**

The second phase of the testing consisted of applying the new algorithm to a set of engineering test problems where known sensitivities and Hessians could not be developed analytically. To determine the accuracy of the sensitivity derivatives returned by the new method, the actual sensitivities for each test problem were developed by reoptimizing the problem over a range of values for each parameter. The results were then fit with either a linear or quadratic curve, depending on the amount of nonlinearity present, and the resulting curve was used to estimate the derivatives. These were then compared with the derivatives obtained from the RQP based algorithm.

This slide describes the three engineering test problems that were used. Complete descriptions are available in Beltracchi (1988).

**Four Bar Slider Crank Problem:** Design a four bar slider crank mechanism to generate a desired coupler path. Four parameters were studied: a movability criteria parameter, two timing parameters, and the y position of a precision point.

**Weld Beam Problem:** Design a welded beam structure for minimum cost. Parameters studied: fixed length of beam, load on beam, yield stress in beam, and allowable shear in weld.

**Corrugated Bulkhead:** Design a ship bulkhead for minimum weight. Parameters studied: change in position of two stringers, and height of the free liquid.

# ENGINEERING TEST PROBLEMS: RESULTS

The following tables report on the accuracy of the parameter sensitivity derivatives returned by the RQP based algorithm for the engineering test problems. The following general conclusions can be drawn from these results:

- The method produces results for  $df/dp$ ,  $dx/dp$ , and  $dg/dp$  that are in the range of 3-4 significant digits of accuracy.
- The results for the four bar mechanism problem are generally worse than the other two problems. This is due to the highly nonlinear nature of this problem and the difficulty in locating accurate optimal points.

## Four Bar Slider Crank

	p 1	p 2	p 3	p 4
$\frac{df}{dp}$	2.0(10 <sup>-5</sup> )	4.86(10 <sup>-3</sup> )	2.25(10 <sup>-5</sup> )	3.29(10 <sup>-7</sup> )
$\epsilon_X$	2.32(10 <sup>-2</sup> )	6.38(10 <sup>-3</sup> )	7.67(10 <sup>-2</sup> )	2.65(10 <sup>-2</sup> )
$\epsilon_U$	3.62(10 <sup>-1</sup> )	3.34(10 <sup>-2</sup> )	6.14(10 <sup>-2</sup> )	5.48(10 <sup>-2</sup> )
$\epsilon_g$	8.81(10 <sup>-5</sup> )	1.94(10 <sup>-3</sup> )	7.82(10 <sup>-2</sup> )	9.36(10 <sup>-3</sup> )

## Welded Beam

	p 1	p 2	p 3	p 4
$\frac{df}{dp}$	1.26(10 <sup>-5</sup> )	8.7(10 <sup>-6</sup> )	8.62(10 <sup>-6</sup> )	1.89(10 <sup>-4</sup> )
$\epsilon_X$	7.59(10 <sup>-5</sup> )	4.28(10 <sup>-5</sup> )	3.12(10 <sup>-5</sup> )	1.26(10 <sup>-4</sup> )
$\epsilon_U$	2.45(10 <sup>-4</sup> )	3.54(10 <sup>-4</sup> )	1.80(10 <sup>-4</sup> )	1.75(10 <sup>-4</sup> )
$\epsilon_g$	8.61(10 <sup>-7</sup> )	1.05(10 <sup>-4</sup> )	1.27(10 <sup>-18</sup> )	1.45(10 <sup>-4</sup> )

## Bulkhead Problem

	p 1	p 2	p 3
$\frac{df}{dp}$	3.21(10 <sup>-5</sup> )	3.34(10 <sup>-6</sup> )	3.51(10 <sup>-5</sup> )
$\epsilon_X$	5.21(10 <sup>-5</sup> )	2.81(10 <sup>-5</sup> )	1.92(10 <sup>-4</sup> )
$\epsilon_U$	7.74(10 <sup>-5</sup> )	1.08(10 <sup>-3</sup> )	8.11(10 <sup>-5</sup> )
$\epsilon_g$	5.12(10 <sup>-5</sup> )	2.69(10 <sup>-5</sup> )	3.23(10 <sup>-5</sup> )

## CONCLUSIONS

In this paper, we have proposed an alternative to current methods for estimating parameter sensitivities. The new method is based on combining the use of an RQP algorithm with differencing formulas which provides a means to estimate the sensitivities without the need for calculating second order derivatives. The method has been tested against two different test sets, one with analytical derivatives available and one without, and in both cases the method was able to accurately determine the sensitivity derivatives. The two major issues in implementing the algorithm concern the ability to formulate an accurate approximation to the Hessian of the Lagrangian and the ability to accurately estimate the modified Kuhn-Tucker sensitivity equations using the differencing formulas. Based on the testing performed so far, we are led to the following conclusions:

1. In terms of efficiency, the method is competitive with existing methods.
2. Parameter sensitivity analysis can be performed using the RQP based method.
3. The Hessian approximation is improved if updating is allowed during the sensitivity calculations.
4. The SR1/PD/BFS update in general provided more accurate estimates of the Hessian of the Lagrangian than either the BFS or SR1 updates on our test set. The initial testing of this update was very encouraging in terms of the convergence of the Hessian approximation to the true Hessian.

1. The method is competitive with existing methods.
2. Parameter sensitivity analysis can be performed using the RQP based method.
3. The Hessian approximation is improved if updating is allowed during the sensitivity calculations.
4. The SR1/PD/BFS update in general provided more accurate estimates of the Hessian of the Lagrangian than either the BFS or SR1 methods.

## REFERENCES

- Arbuckle, P. D., and Sliwa, S. M., 1984, "Experiences Performing Conceptual Design Optimization of Transport Aircraft" in *Recent Experiences in Multidisciplinary Analysis and Optimization, Part 1*, Compiled by J. Sobieski, NASA CP 2327, pp 87-101.
- Armacost, R. L., and Fiacco, A. V., 1974, "Computational Experience in Sensitivity Analysis for Nonlinear Programming", in *Mathematical Programming*, Vol. 6, pp 301-326.
- Arora, J. S., and Tseng, C. H., 1987, "Discussion on ASME Paper No. 86-DET-26," in *ASME Journal of Mechanisms, Transmissions and Automation in Design*, June, Vol 109 No. 2.
- Barthelemy, J. F., and Sobieszczanski-Sobieski, J., 1983 a, "Extrapolation of Optimum Designs Based of Sensitivity Derivatives," in *AIAA Journal*, Vol. 21, May , pp 797-799.
- Barthelemy, J. F., and Sobieszczanski-Sobieski, J., 1983 b, "Optimum Sensitivity Derivatives of Objective Functions in Nonlinear Programming" in *AIAA Journal*, Vol. 21, June pp 913-915.
- Bartholomew-Biggs, M. C., 1987, "Recursive Quadratic Programming Methods Based on the Augmented Lagrangian," in *Mathematical Programming Study 31, Computational Mathematical Programming*, edited by K. L. Hoffman, R. H. F. Jackson, and J. Telgen, North Holland - Amsterdam, pp 21-41.
- Beltracchi, T. J., 1985, *An Investigation of Pshenichnyi's Recursive Quadratic Programming Technique for Engineering Optimization*, MS Thesis Rensselaer Polytechnic Institute, Dec. 1985
- Beltracchi, T. J., and Gabriele, G. A., 1987, "An Investigation of Pshenichnyi's Recursive Quadratic Programming Technique for Engineering Optimization", in *ASME Journal of Mechanisms, Transmissions, and Automation in Design*, Vol 109 No. 2 pp 248-256 June 1987.
- Beltracchi, T. J., and Gabriele, G. A., "An Investigation of New Methods for Estimating Parameter Sensitivities," NASA CR 183195.
- Beltracchi, T. J., "An Investigation of Parameter Sensitivity Analysis by the Recursive Quadratic Programming Method," Ph.D. Thesis, Rensselaer Polytechnic Institute, 1988.
- Buys, J. D. and Gonin, R., 1977 , "The use of Augmented Lagrangian Functions for Sensitivity Analysis in Nonlinear Programming." in *Mathematical Programming*, Vol. 12 No. 2, pp 281-284.
- Cha, J. Z., and Mayne R. W. 1987, "Optimization with Discrete Variables Via Recursive Quadratic Programming: Part II: Algorithm and Results," in *ASME DE Vol. 10-1, The Proceedings of the 1987 Design Automation Conference*, Boston Ma.
- Cullum, J. and Brayton, R. K., 1979 "Some Remarks on the Symmetric Rank-One Update," in *Journal of Optimization Theory and Applications*, Vol 29, No 4., pp 493-519 December
- Dantzig, G. B., 1963, *Linear Programming and Extensions*, Princeton University Press, Princeton New Jersey.
- Dennis, J. E., and Schnable, R. B., 1983, *Numerical Methods for Unconstrained Optimization and Nonlinear Equations*, Prentice Hall.

Falk, J. E., and Fiacco, A. V., 1982, "The use of Mathematical Programming: Who let the Man Out.", in *Computers and Operations Research*, Vol. 9, No. 1, pp. 3-5.

Fiacco, A. V., 1976, "Sensitivity Analysis For Nonlinear Programming Using Penalty Methods", in *Mathematical Programming*, Vol. 10, pp 287-311.

Fiacco, A. V., 1983, *Introduction to Sensitivity and Stability Analysis in Nonlinear Programming*, Academic Press, New York.

Fiacco, A. V., and Ghaemi, A., 1982, "Sensitivity Analysis of a Nonlinear Structural Design Problem", in *Computers and Operations Research*, Vol. 9, No. 1, pp 29-55.

Gal, T., 1984, "Linear Parametric Programming- A Brief Survey," in *Mathematical Programming Study 21*, North Holland, pp 43-68.

Gill, P. E., Murray, W., Saunders, M. A., and Wright, M. H., 1983 "Computing Forward-Difference Intervals for Numerical Optimization", in *SIAM Journal of Scientific and Statistical Computing*, Vol. 4, No. 2, June 1983, pp 310-321.

Gill, P. E., Murray, W., Saunders, M. A., and Wright, M. H., 1986, "Considerations of Numerical Analysis in a Sequential Quadratic Programming Method", in *Numerical Analysis Lecture Notes in Mathematics No. 1230*, Springer Verlag, New York 1986.

Haftka, R. T., Iott, J., and Adelman, H. M., 1985, "Selecting step sizes in sensitivity analysis by finite differences," Aug. 1985, NASA TM 86382.

Jittorntrum, K., 1984, "Solution Point Differentiability without Strict Complementarity in Nonlinear Programming," in *Mathematical Programming Study 21*, North Holland Amsterdam pp 127-138.

McKeown, J. J., 1980, "Parametric Sensitivity Analysis of Nonlinear Programming Problems", In *Nonlinear Optimization Theory and Algorithms*, L. C. W. Dixon, E Spedicato and G. P. Szego editors, North Holland, Amsterdam, pp. 387-406

Robertson, W. D., and Gabriele, G. A., 1987, "The Optimal Design of a Rotary Type Actuator for Magnetic Disk Drives," in *Advances in Design Automation - 1987, Volume One: Design Methods, Computer Graphics, and Expert Systems*, ASME DE-Vol. 10-1, pp 107-114, to Appear in *ASME Journal of Mechanisms, Transmissions, and Automation in Design*.

Sandgren E., 1977, "The Utility of Nonlinear Programming Algorithms", Ph.D. dissertation Purdue University Dec. 1977, West Lafayette In.

Schmit, L. A., and Chang, K. J., 1984, "Optimum Design Sensitivity Based on Approximation Concepts and Dual Methods", in *International Journal for Numerical Methods in Engineering*, vol. 20, pp 39-75.

Sobieszczanski-Sobieski, J., Barthelemy, J. F., and Riley K. M., 1981, "Sensitivity of Optimum Solutions to Problem Parameters", *Proceedings of the AIAA/ASME/ASCE/AHS 22nd Structures, Structural Dynamics and Materials Conference*, Atlanta, Ga., April 1981, pp 184-205, also NASA TM 83134, May 1981, *AIAA Journal*, Vol 20., Sept. 1982, pp. 1291-1299.

Sobieszczanski-Sobieski, J., James, B. B., and Riley, M. F., 1987, "Structural Sizing by Generalized, Multilevel Optimization," in *AIAA Journal* Vol. 25, No. 1, January 1987, pp 139-145

Vanderplaats, G. N., 1984, "An Efficient Feasible Directions Algorithm for Design Synthesis", in AIAA Journal, Vol. 22, No. 11, Nov. , pp. 1633-1639

Vanderplaats, G. N., and Cai, H. D., 1987, "Alternative Methods for Calculating Sensitivity of Optimized Designs to Problem Parameters," in Sensitivity Analysis in Engineering, Edited by H. M. Alderman and R. T. Haftka, NASA CP-2457, pp 19-32.

Vanderplaats, G. N., and Yoshida, N., 1985, "Efficient Calculation of Optimum Design Sensitivity" in AIAA Journal, Nov. , pp 1798-1803.